

## CLAIMS

We claim:

1. In a computerized system environment including computer-executable instructions, and one or more interfaces for accessing the computer-executable instructions, a method of testing the computer-executable instructions through each of the one or more interfaces using a single testing program, the method comprising the acts of:

identifying one or more interfaces that are intended to access an identified application program;

identifying an application program interface that is common to each of the one or more interfaces that can access the application program, such that a function of the application program that can be accessed by each of the one or more interfaces can be tested;

providing at least one representation of a first value to the application program through the common application program interface; and

receiving a result from the application program.

2. The method as recited in claim 1, wherein the at least one representation of the first value is unique to at least one of the one or more interfaces.

3. The method as recited in claim 2, wherein the at least one representation of the first value is identified automatically prior to providing the at least one representation to the application program.

4. The method as recited in claim 1, wherein the one or more interfaces includes one or more of a telephone user interface, a graphical user interface, a command-line interface, and a machine-based interface.
5. The method as recited in claim 1, wherein the identified application program is an application program to be tested.
6. The method as recited in claim 1, further comprising generating a test program that is configured to access the identified application program through the identified common application program interface.
7. The method as recited in claim 6, wherein the first value is provided to the application program by the test program through the identified common application program interface.
8. The method as recited in claim 6, further comprising identifying one or more other application program interfaces that are common to the identified user interfaces.
9. The method as recited in claim 8, further comprising converting the test program such that it is configured to access the identified application program through at least one of the one or more other application program interfaces.
10. The method as recited in claim 1, further comprising receiving one or more results from the application program through the corresponding one or more interfaces that are intended to access the application program.
11. The method as recited in claim 10, further comprising, based on the received one or more results, identifying an expected result by which the received one or more results can be compared.

12. In a computerized system environment including computer-executable instructions, and one or more interfaces for accessing the computer-executable instructions, a method of testing the computer-executable instructions through each of the one or more interfaces using a single testing program, the method comprising:

an act of identifying an application program to be tested;

an act of identifying one or more interfaces that are intended to access the application program; and

a step for determining the functionality of the application program with the one or more interfaces by using a single testing program that incorporates an application program interface that is common to each of the one or more interfaces.

13. The method as recited in claim 12, wherein the step for determining the functionality of the application program with the one or more interfaces comprises corresponding the acts of:

identifying an application program interface that is common to each of the one or more interfaces that can access the application program, such that a function of the application program that can be accessed by each of the one or more interfaces can be tested;

providing at least one representation of a first value to the application program through the common application program interface; and

receiving a result from the application program.

14. In a computerized system environment including computer-executable instructions, and one or more interfaces for accessing the computer-executable instructions, a method of testing an application program through each of the one or more interfaces using a single testing program, the method comprising:

identifying a plurality of interfaces that are intended to access an application program;

sending a first value to the application program for each of the plurality of identified interfaces, wherein the first value is sent using an application program interface that is common to each of the plurality of identified interfaces; and

receiving a plurality of results from the application program, wherein each result in the plurality corresponds to an identified one of the plurality of interfaces; and

comparing the plurality of results to identify an expected result.

15. The method as recited in claim 14, further comprising sending a next value to the application program for each of the plurality of identified interfaces.

16. The method as recited in claim 15, further comprising receiving a next result from the application program that is based in part on the next value that has been sent to the application.

17. The method as recited in claim 16, further identifying that the application is interoperable with at least one of the identified interfaces by comparing the next result with the expected result.

18. The method as recited in claim 14, further comprising generating a test program that is configured to access the application program through the identified common application program interface.

19. The method as recited in claim 18, further comprising identifying one or more other application program interfaces that are common to the identified user interfaces.

20. The method as recited in claim 19, further comprising converting the test program such that it is configured to access the identified application program through at least one of the one or more other application program interfaces.

21. In a computerized system environment including computer-executable instructions, and one or more interfaces for accessing the computer-executable instructions, a computer program product having computer-executable code stored thereon that, when executed, causes a computerized system to perform a method of testing an application program through each of the one or more interfaces using a single testing program, the method comprising:

identifying one or more interfaces that are intended to access an identified application program;

identifying an application program interface that is common to each of the one or more interfaces that can access the application program, such that a function of the application program that can be accessed by each of the one or more interfaces can be tested;

providing at least one representation of a first value to the application program through the common application program interface; and

receiving a result from the application program.

22. In a computerized system environment including computer-executable instructions, and one or more interfaces for accessing the computer-executable instructions, a computer program product having computer-executable instructions stored thereon that, when executed, cause a computerized system to perform a method of testing an application program through each of the one or more interfaces using a single testing program, the method comprising:

identifying a plurality of interfaces that are intended to access an application program;

sending a first value to the application program for each of the plurality of identified interfaces, wherein the first value is sent using an application program interface that is common to each of the plurality of identified interfaces; and

receiving a plurality of results from the application program, wherein each result in the plurality corresponds to an identified one of the plurality of interfaces; and

comparing the plurality of results to identify an expected result.